Filtragem

Filtros lineares

- Convolução
  - - Filtro passa-baixas (suavização)
  - - Filtro passa-altas (realce)
  - - Filtros direcionais

Filtros não lineares

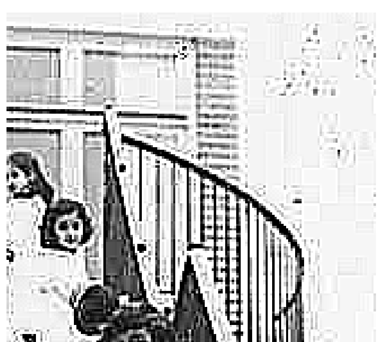
  - Mediana
  - Gradiente

Processamento de imagens

Prof. Dr. Jorge Centeno

2024

Para que servem?



Filtros lineares

Os filtros lineares resultam da *convolução* de uma janela móvel e a imagem.

A Convolução em uma dimensão:

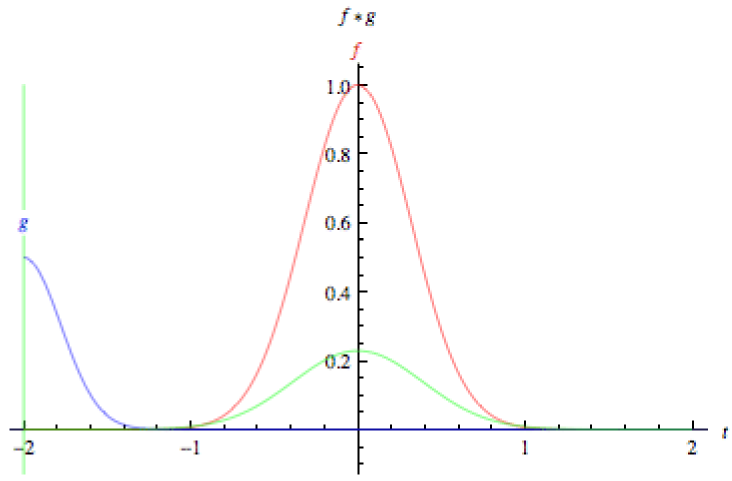
A convolução é um operador **linear** que combina matematicamente duas funções previamente definidas para gerar uma terceira função (resultado) definida ao longo do domínio das funções de entrada.

O operador da convolução mede a **soma do produto** das funções de entrada ao longo do domínio, definido pela superposição delas, deslocando uma delas em relação à outra.

$$G(x) = \Sigma\ F(x) * P(x+dx)$$

- X denota o domínio das funções
- dx denota o deslocamento de P neste domínio

Ao deslocar a segunda função e multiplicar pela primeira, são obtidos valores diferentes em função do deslocamento.

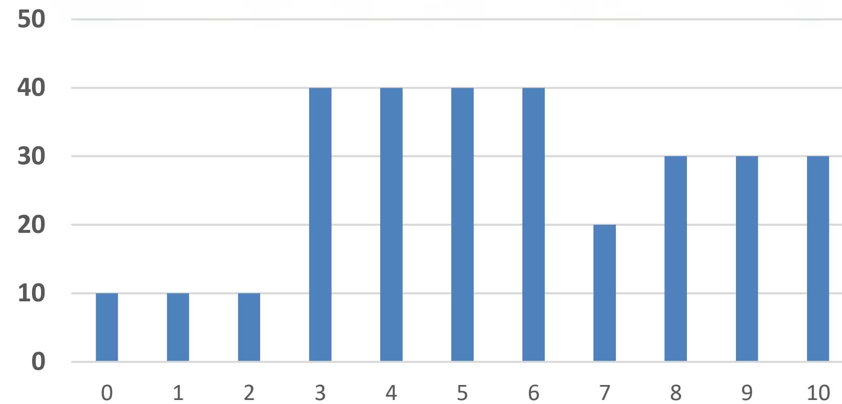


Duas propriedades

- A função que se desloca pode funcionar como um "filtro" que modifica a primeira função
- Variando a função que se desloca podem ser obtidos diferentes efeitos *filtros"
- Estas transformações são lineares

Exemplo DISCRETO

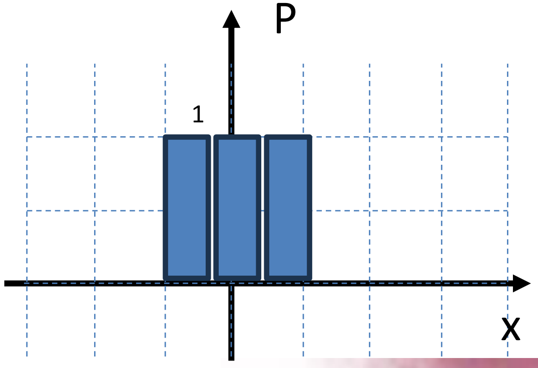
- Domínio discreto:
- F(x) de 0 a 10 (inteiros)



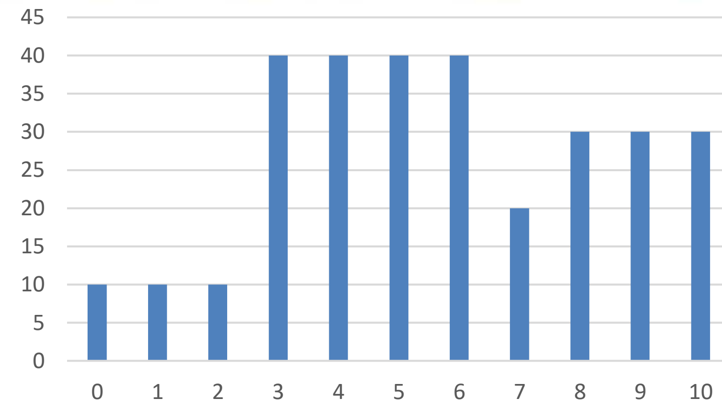
X	0	1	2	3	4	5	6	7	8	9	10
F	10	10	10	40	40	40	40	20	30	30	30

- $$P(x) = \begin{Bmatrix} P(-1) = 1 \\ P(0) = 1 \\ P(1) = 1 \end{Bmatrix}$$

X	-1	0	1
|P	1	1	1



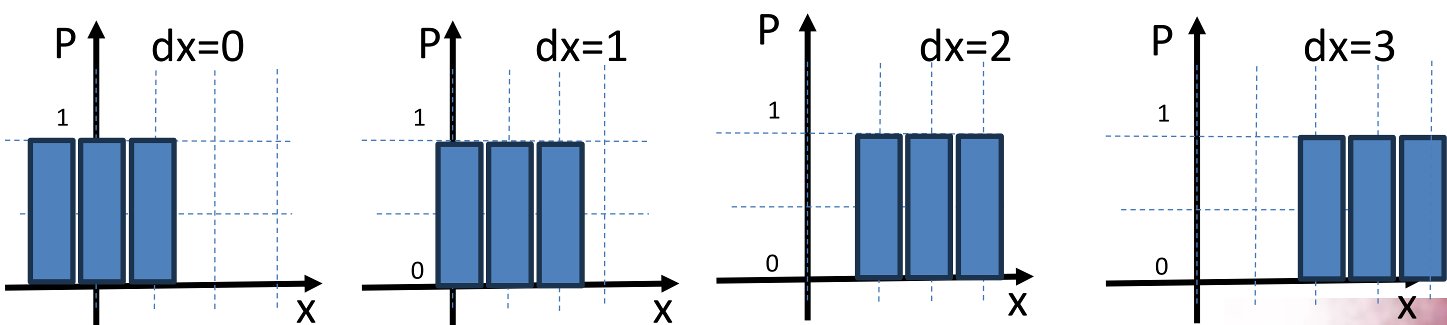
exemplo



$$G(x) = \Sigma \; F(x) * P(x+dx)$$

- X denota o domínio das funções
- dx denota o deslocamento de P neste domínio

Então, para cada possível “dx” deve-se calcular a soma da multiplicação das duas funções...



Convolução

X	0	1	2	3	4	5	6	7	8	9	10
F	10	10	10	40	40	40	40	20	30	30	30

1	1	1

dx=0 não se pode calcular a convolução porque não F não está definida para x=-1

Se dx=1

x	0	1	2	3	4	5	6	7	8	9	10
P	10	10	10	40	40	40	40	20	30	30	30
F	1	1	1								
G		30									

deslocar

começando em 0

dx=1

x	0	1	2	3	4	5	6	7	8	9	10
P	10	10	10	40	40	40	40	20	30	30	30
F	1	1	1	0	0	0	0	0	0	0	0

dx=2

x	0	1	2	3	4	5	6	7	8	9	10
P	10	10	10	40	40	40	40	20	30	30	30
F	0	1	1	1	0	0	0	0	0	0	0

dx=3

x	0	1	2	3	4	5	6	7	8	9	10
P	10	10	10	40	40	40	40	20	30	30	30
F	0	0	1	1	1	0	0	0	0	0	0

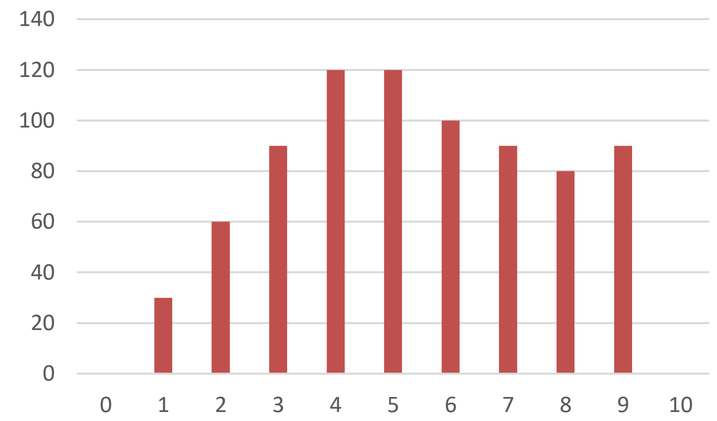
dx=4

x	0	1	2	3	4	5	6	7	8	9	10
P	10	10	10	40	40	40	40	20	30	30	30
F	0	0	0	1	1	1	0	0	0	0	0

…

dx=9

x	0	1	2	3	4	5	6	7	8	9	10
P	10	10	10	40	40	40	40	20	30	30	30
F	0	0	0	0	0	0	0	0	1	1	1



0	30	60	90	120	120	100	90	80	90	0

Filtros lineares

Em processamento de imagens, os filtros lineares resultam da convolução de uma janela móvel e a imagem no espaço 2-D.

O resultado de um filtro linear pode ser escrito na seguinte forma:

$$G(y,x) = \Sigma\ \Sigma\ (\ F(x,y) * P(x+dx, y+dy)\ )$$

onde

- y,x representam as coordenadas do pixel e
- dx, dy o deslocamento relativo
- P(i,j) representa o filtro
- G(y,x) é a imagem resultante, filtrada

dx=1,dy=1

1	1	1	1	2	1	1	0
1	1	2	1	3	1	1	1
1	1	3	9	2	1	1	0
1	1	3	1	2	1	1	1
9	7	8	9	8	9	9	8
9	9	8	8	9	9	9	8
9	9	9	8	9	8	9	9
9	8	9	8	8	9	9	9

dx=1,dy=2

1	1	1	1	2	1	1	0
1	1	2	1	3	1	1	1
1	1	3	9	2	1	1	0
1	1	3	1	2	1	1	1
9	7	8	9	8	9	9	8
9	9	8	8	9	9	9	8
9	9	9	8	9	8	9	9
9	8	9	8	8	9	9	9

dx=1,dy=3

1	1	1	1	2	1	1	0
1	1	2	1	3	1	1	1
1	1	3	9	2	1	1	0
1	1	3	1	2	1	1	1
9	7	8	9	8	9	9	8
9	9	8	8	9	9	9	8
9	9	9	8	9	8	9	9
9	8	9	8	8	9	9	9

dx=1,dy=4

1	1	1	1	2	1	1	0
1	1	2	1	3	1	1	1
1	1	3	9	2	1	1	0
1	1	3	1	2	1	1	1
9	7	8	9	8	9	9	8
9	9	8	8	9	9	9	8
9	9	9	8	9	8	9	9
9	8	9	8	8	9	9	9

dx=1,dy=5

1	1	1	1	2	1	1	0
1	1	2	1	3	1	1	1
1	1	3	9	2	1	1	0
1	1	3	1	2	1	1	1
9	7	8	9	8	9	9	8
9	9	8	8	9	9	9	8
9	9	9	8	9	8	9	9
9	8	9	8	8	9	9	9

dx=1,dy=6

1	1	1	1	2	1	1	0
1	1	2	1	3	1	1	1
1	1	3	9	2	1	1	0
1	1	3	1	2	1	1	1
9	7	8	9	8	9	9	8
9	9	8	8	9	9	9	8
9	9	9	8	9	8	9	9
9	8	9	8	8	9	9	9

dx=2,dy=1

1	1	1	1	2	1	1	0
1	1	2	1	3	1	1	1
1	1	3	9	2	1	1	0
1	1	3	1	2	1	1	1
9	7	8	9	8	9	9	8
9	9	8	8	9	9	9	8
9	9	9	8	9	8	9	9
9	8	9	8	8	9	9	9

dx=2,dy=2

1	1	1	1	2	1	1	0
1	1	2	1	3	1	1	1
1	1	3	9	2	1	1	0
1	1	3	1	2	1	1	1
9	7	8	9	8	9	9	8
9	9	8	8	9	9	9	8
9	9	9	8	9	8	9	9
9	8	9	8	8	9	9	9

3x3

Para um pixel na posição i,j, deve-se calcular o valor digital médio dele e seus oito vizinhos

- Os vizinhos são

i-1,j-1	i-1, j	i-1,j+1
i ,j-1	i , j	i , j+1
i+1,j-1	i+1, j	i+1,j+1

```
1 1 1 1 2 1 1 0
1 1 2 1 3 1 1 1
1 1 3 9 2 1 1 0
1 1 3 1 2 1 1 1
9 7 8 9 8 9 9 8
9 9 8 8 9 9 9 8
9 9 9 8 9 8 9 9
9 8 9 8 8 9 9 9
```

Variando um índice em colunas j+dc dc=[-1,0, 1]
E variando um índice de linhas i+dl dl=[-1, 0,1]

[i+dl, j+dc]

Da definição de filtro, entende-se que:

- Um resultado pode ser calculado para cada posição da janela móvel dentro da imagem.
- O resultado é sempre atribuído ao pixel central.

Este novo valor é calculado somando o produto do pixel na janela 3x3 e o respectivo peso, definido no filtro.

filtros lineares

Os filtros lineares são definidos em termos de seu efeito na imagem

- *passa baixas*
- *passa altas*
- *direcionais*

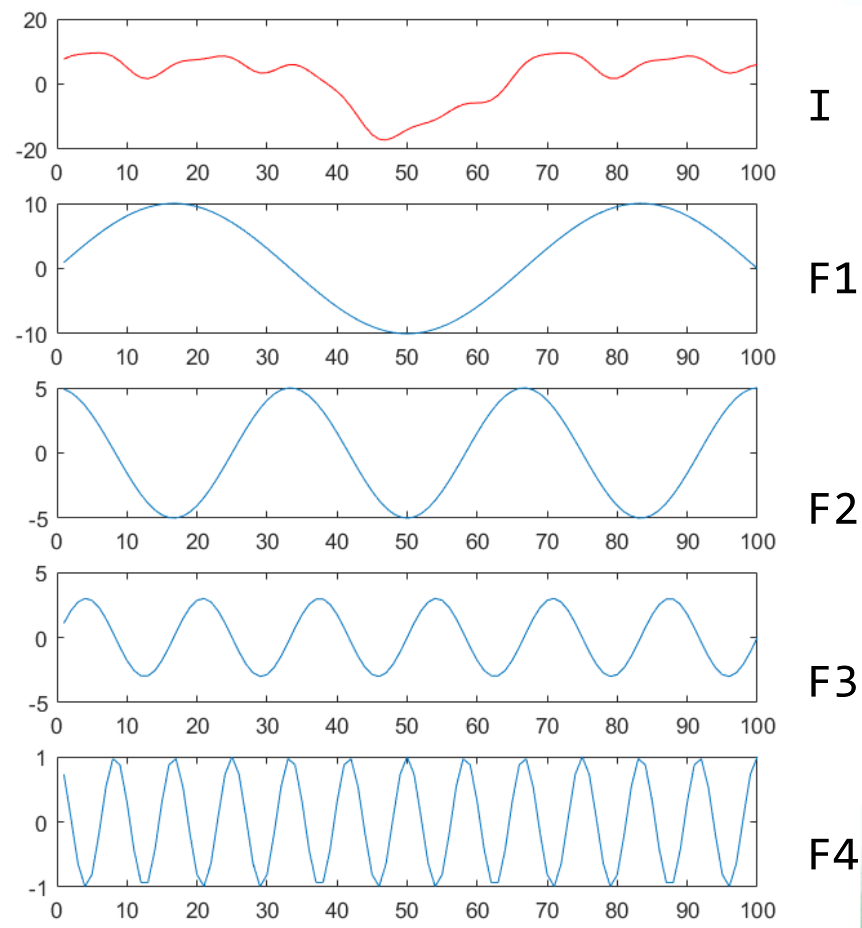
- FILTRO PASSA-BAIXAS
- LOW-PASS FILTER
- Suavização

Conceito de Frequencias

Uma função pode ser descrita como a soma de várias funções senoidais com diferentes frequências.

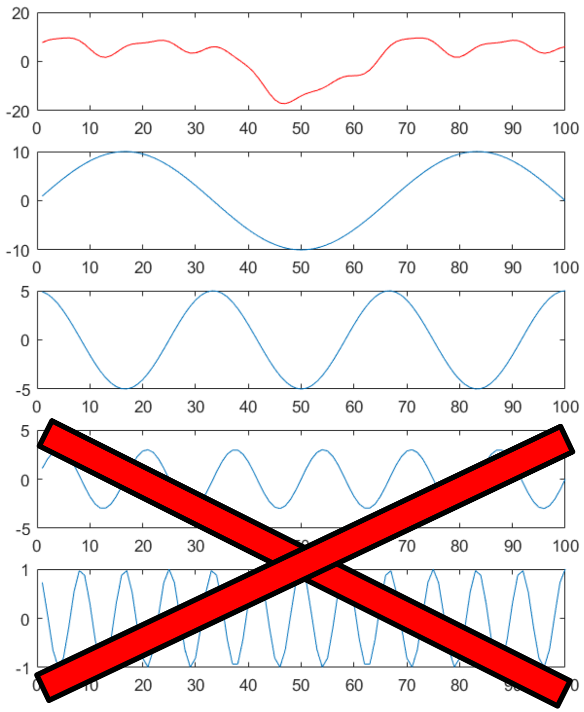
I=F1+F2+F3+F4 + ...

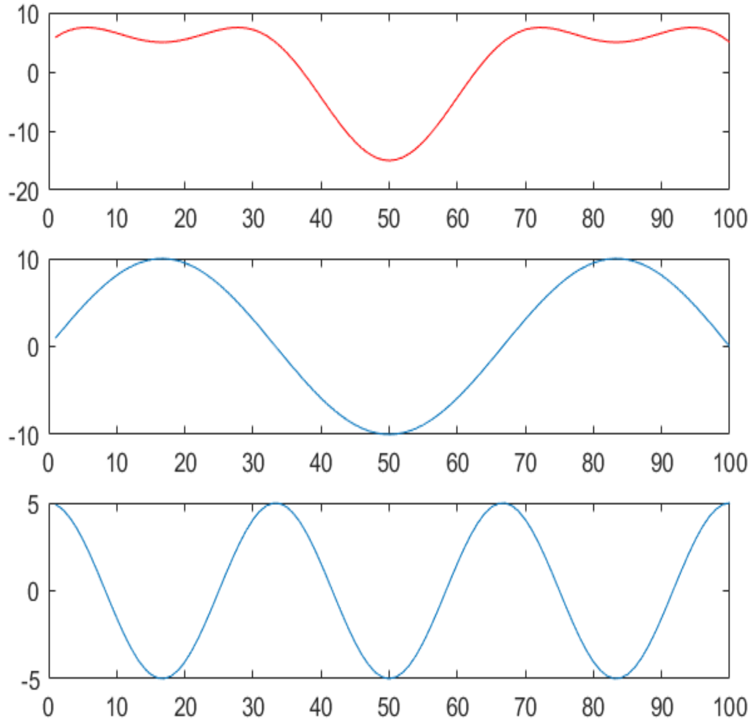
As funções com menores frequencias são responsáveis por dar a forma geral da curva, enquanto as com menores frequencias representam os detalhes



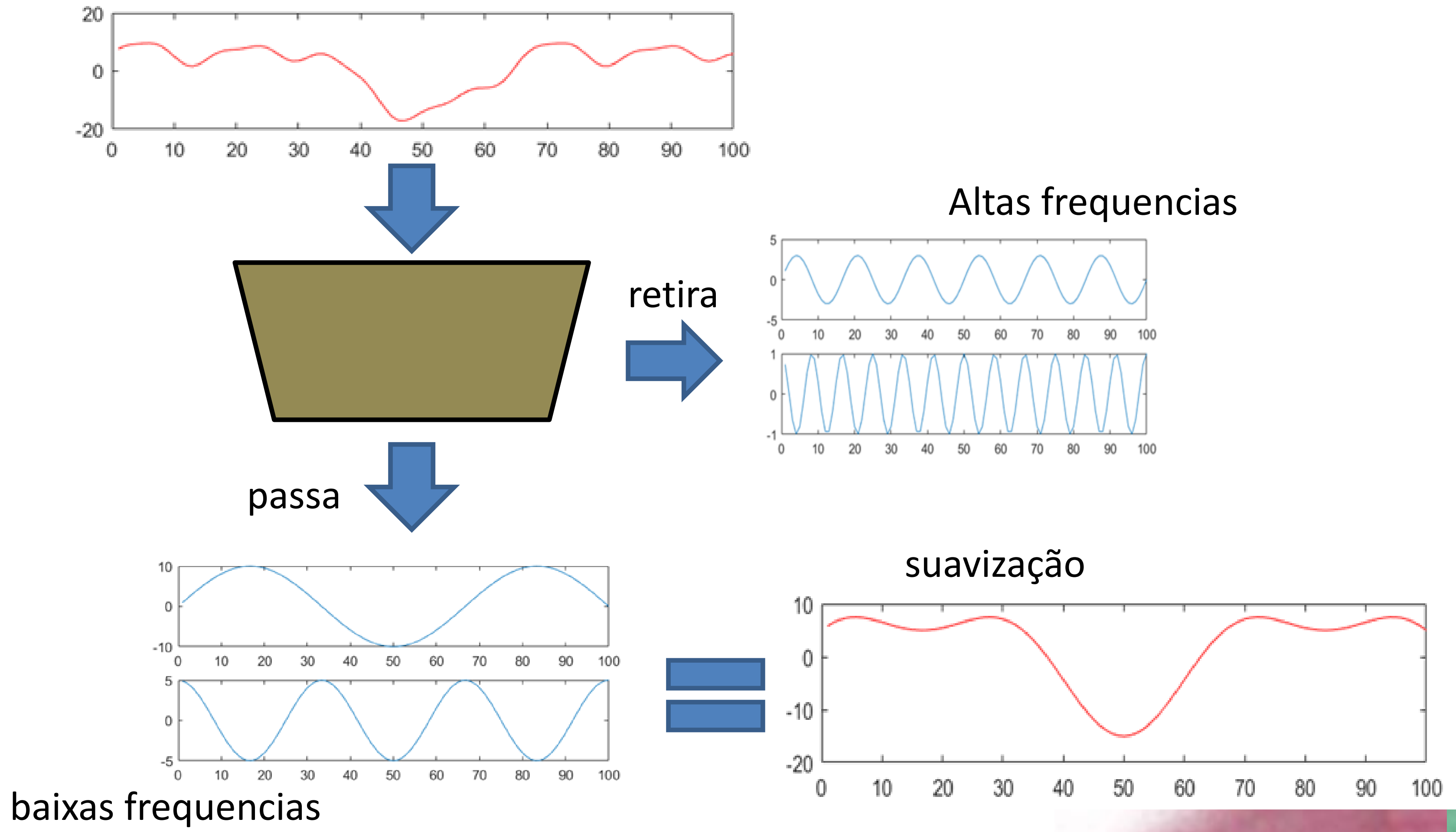
filtragem

O que ocorre se retiramos as altas frequencias?





Filtragem seletiva de frequencias



Filtro passa-baixas (suavização)

- Atenua as altas frequências, associadas a detalhes na imagem, e deixa apenas as baixas frequências.
- O efeito é a remoção de detalhes da imagem e sua suavização. A imagem filtrada apresenta uma aparência de “fora de foco”, e as áreas presentes na imagem tornam-se mais homogêneas.
- O efeito é atingido substituindo o pixel central pela média da janela. A média pode ser uma média simples ou uma média ponderada, onde diferentes pesos são atribuídos aos vizinhos em função de sua proximidade ao pixel central.
- Exemplo:

1	1	1	1	2	1	1	0
1	1	2	1	3	1	1	1
1	1	3	9	2	1	1	0
1	1	3	1	2	1	1	1
9	7	8	9	8	9	9	8
9	9	8	8	9	9	9	8
9	9	9	8	9	8	9	9
9	8	9	8	8	9	9	9

Filtro (3x3)

1	**1**	**1**
1	**1**	**1**
1	**1**	**1**

Exemplos de filtros passa-baixas

1	1	1
1	1	1
1	1	1

* 1/9

1	2	1
2	4	2
1	2	1

* 1/16

1	1	1
1	4	1
1	1	1

* 1/12

1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1

* 1/25

Filtros de média ponderada.
Para o cálculo do valor final, o resultado da multiplicação dos pesos e os valores da imagem deve ser dividido pela soma dos pesos.

Passa baixas

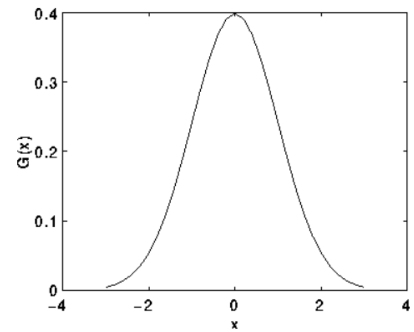
- Original　　　　　　　　passa-baixas (suavização)

FILTRO GAUSSIANO

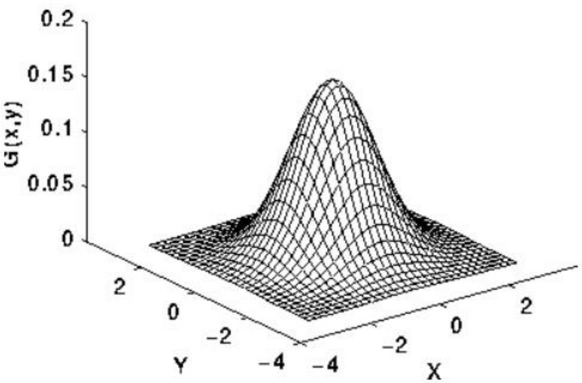
Filtro **passa-baixas** que usa uma função Gaussiana para calcular os pesos do filtro.

Maior peso é dado ao central e o peso diminui com a distância ao pixel central da janela.

$$G(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}}$$



$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

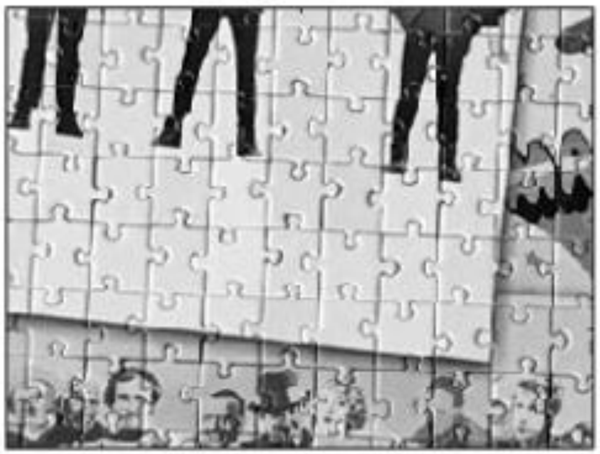


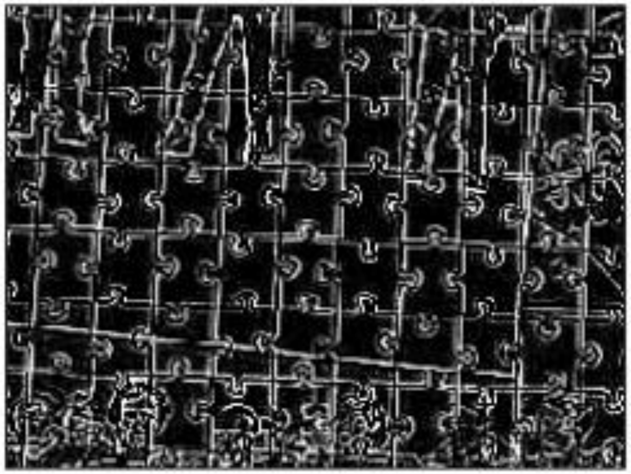
$\frac{1}{273}$

1	4	7	4	1
4	16	26	16	4
7	26	41	26	7
4	16	26	16	4
1	4	7	4	1

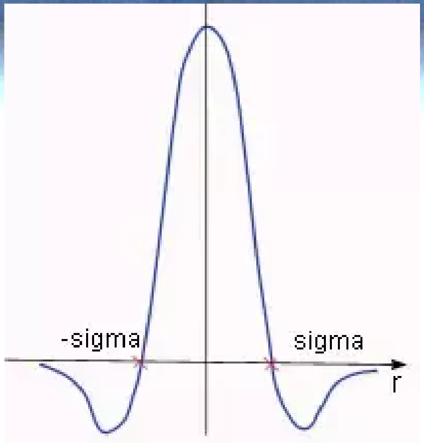
FILTRO LAPLACIANO

- FILTRO LAPLACIANO
- Laplacian Filter





Laplaciano



O filtro **Laplaciano** é um operador que calcula a derivada isotrópica (não depende da direção, em tidas as direções)

0	-1	0
-1	4	-1
0	-1	0

Equivale a calcular o Gradiente local (diferença entre o vizinho e o central) em todas as direções e somar estes gradientes.

Exemplo:

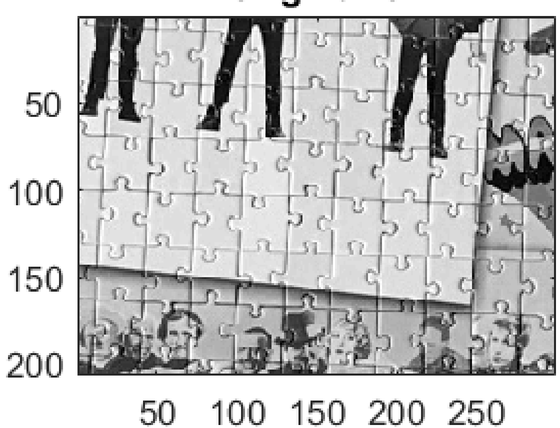
-1	-1	-1
-1	8	-1
-1	-1	-1

Diferença entre o central e seus vizinhos

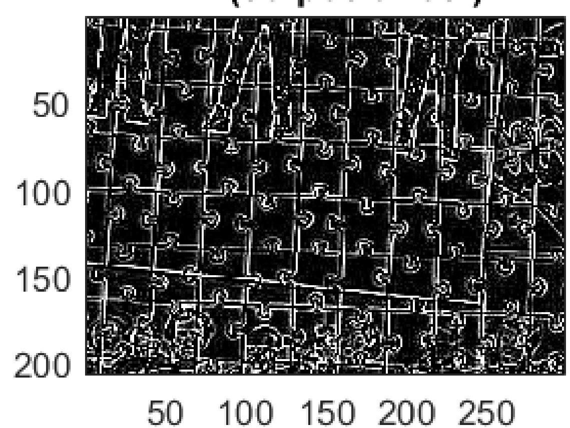
$$
\begin{array}{ccc}
\begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} &
\begin{bmatrix} 0 & -1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} &
\begin{bmatrix} 0 & 0 & -1 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \\
\begin{bmatrix} 0 & 0 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} &
\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} &
\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & -1 \\ 0 & 0 & 0 \end{bmatrix} \\
\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \end{bmatrix} &
\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -1 & 0 \end{bmatrix} &
\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix}
\end{array}
\Rightarrow
\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}
$$

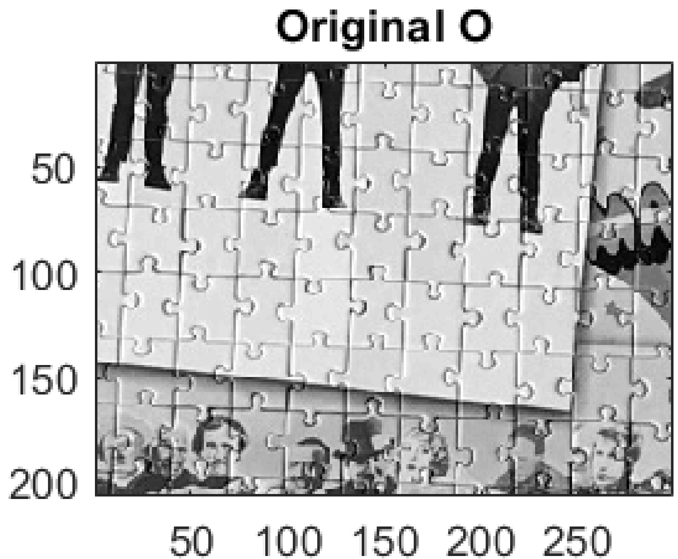
A diferença entre o central e seus oito vizinhos

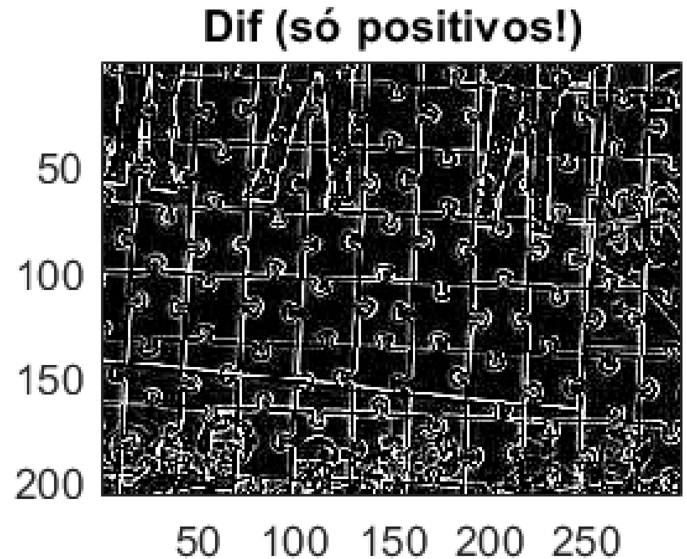
Original O

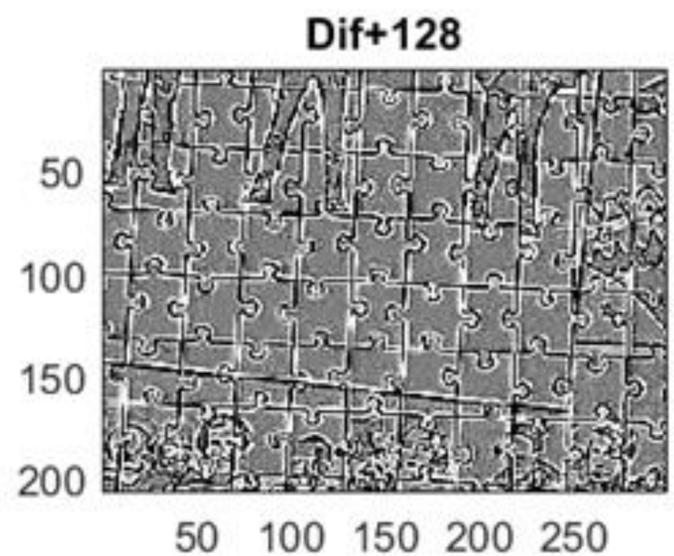


Dif (só positivos!)




Original O
50
100
150
200
50 100 150 200 250

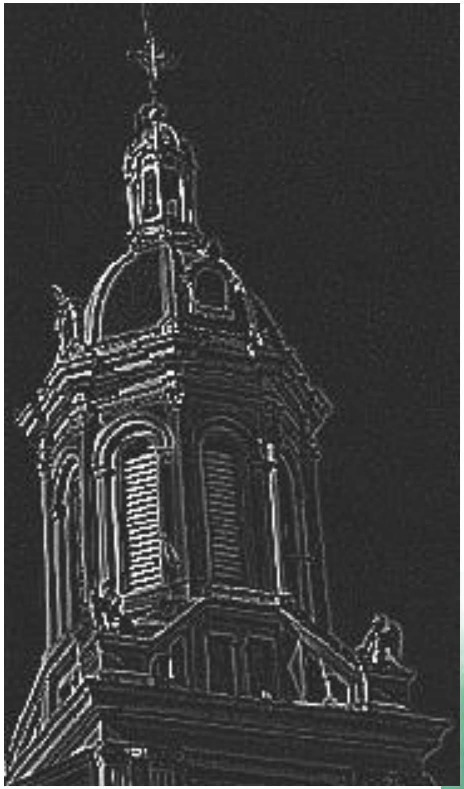

Dif (só positivos!)
50
100
150
200
50 100 150 200 250


Dif+128
50
100
150
200
50 100 150 200 250

- FILTRPPASSA-ALTA
- HIGH-PASS FILTER
- REALCE

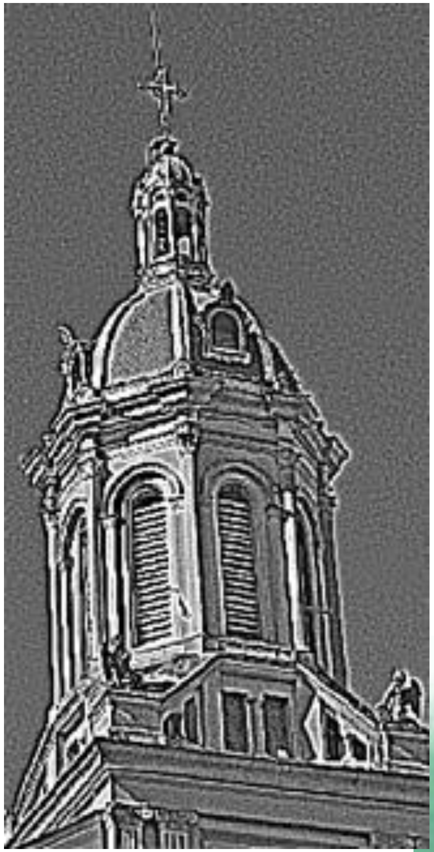
Filtro passa-altas (realce)

- Enfatiza contrastes, realçando os detalhes.
- O nome do filtro explica seu funcionamento, pois nesta transformação as baixas frequências são eliminadas, sendo as altas frequências as únicas remanescentes.
- As altas frequencias estão associadas ao detalhe espacial
- O filtro Laplaciano encontra as diferenças (High-pass)



Filtro passa-altas (realce de detalhes)

- As bordas realçadas podem ser superpostas à umagem original para reforças as bordas e detalhes.
- Este efeito pode ser atingido adicionando à imagem original a diferença entre a imagem original e o resultado de um filtro-passa baixas.
- O resultado da operação é nulo em regiões homogêneas. Em regiões com detalhes, o valor resultante é alto, em função do contraste entre o pixel central e a vizinhança.



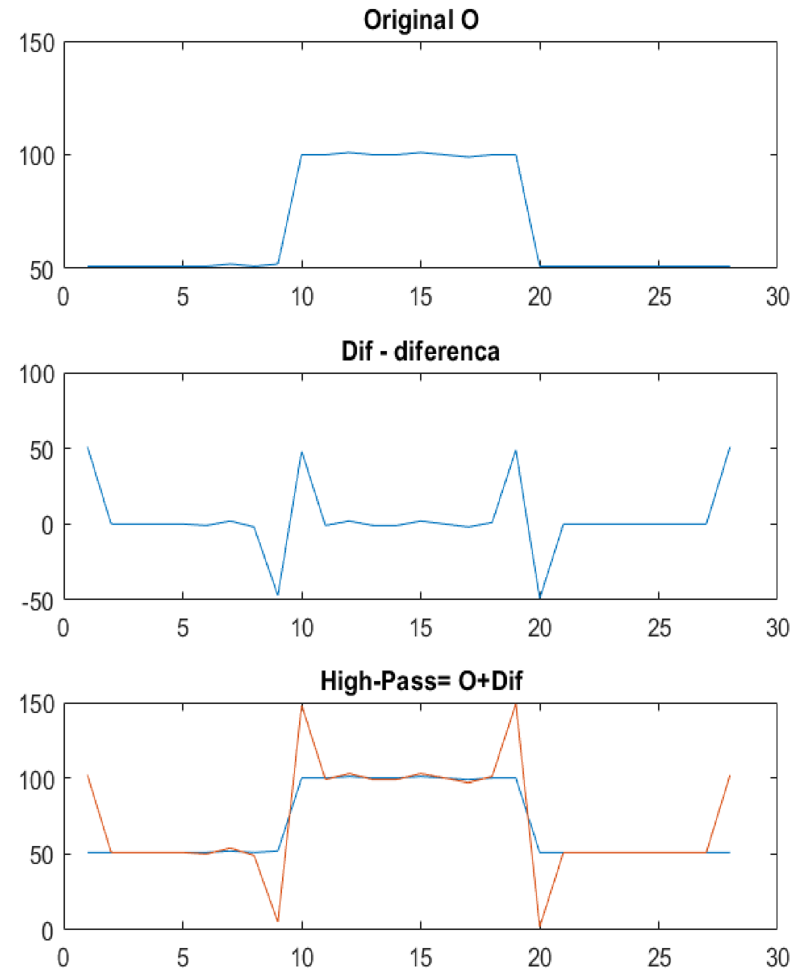
conceito

Para entender... (1D)

Calcule a diferença entre o seguinte e seus vizinhos
Soma= -Y(i-1) + 2*Y(i) -Y(i+1)

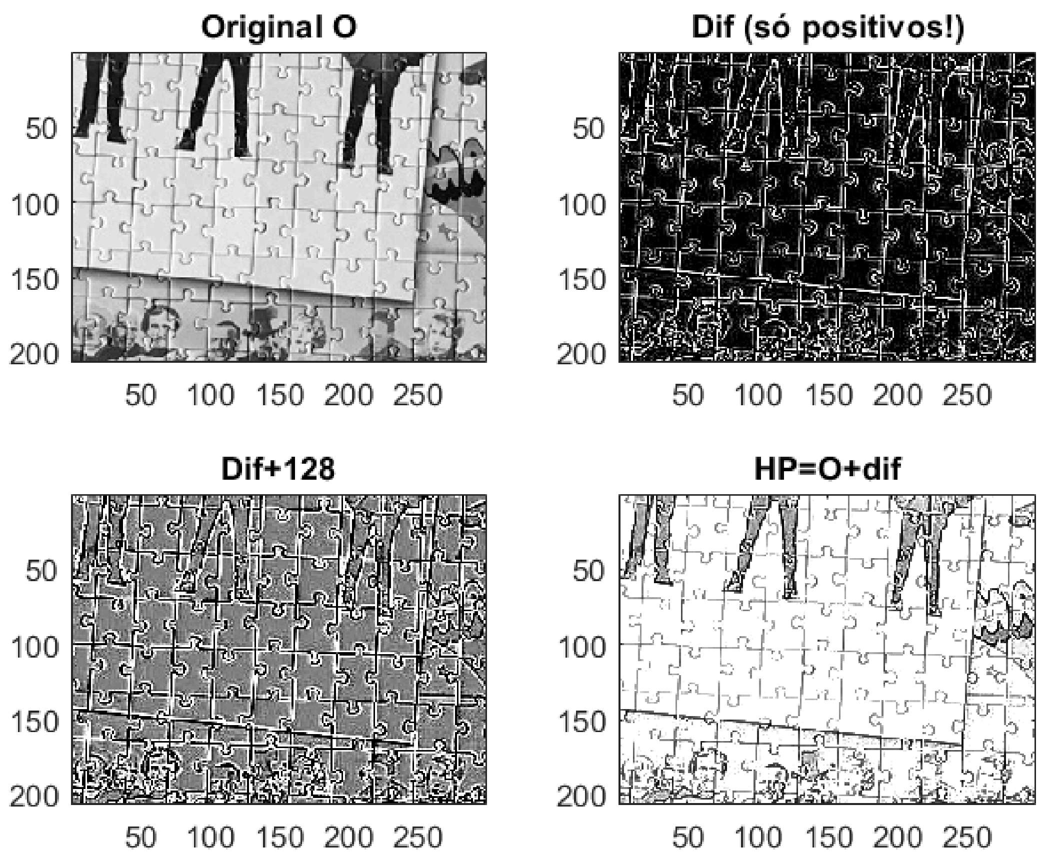
A diferença é
- nula em regiões uniformes;
- alta se Y(i+1)> Y(i) (transição);
- baixa se Y(i+1)<Y(i) (transição).

Somando esta diferença ao valor original se salienta o contraste nas regiões de bordas.
Em regiões uniformes, não ocorre alteração.



Diferença + Central

$$\begin{matrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{matrix} \quad + \quad \begin{matrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{matrix} \quad \rightarrow \quad \begin{matrix} -1 & -1 & -1 \\ -1 & 9 & -1 \\ -1 & -1 & -1 \end{matrix}$$



Somando a diferença ao valor central acentua o contraste

Exemplos de passa-altas

-1	-1	-1
-1	9	-1
-1	-1	-1

0	-1	0
-1	5	-1
0	-1	0

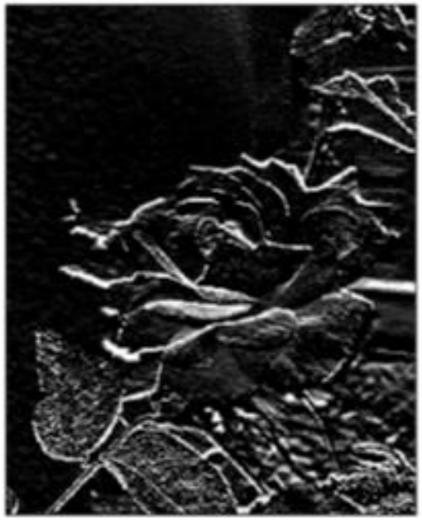
-1	-1	-1	-1	-1
-1	-1	-1	-1	-1
-1	-1	25	-1	-1
-1	-1	-1	-1	-1
-1	-1	-1	-1	-1

Passa altas

•Original passa-altas

Filtros direcionais

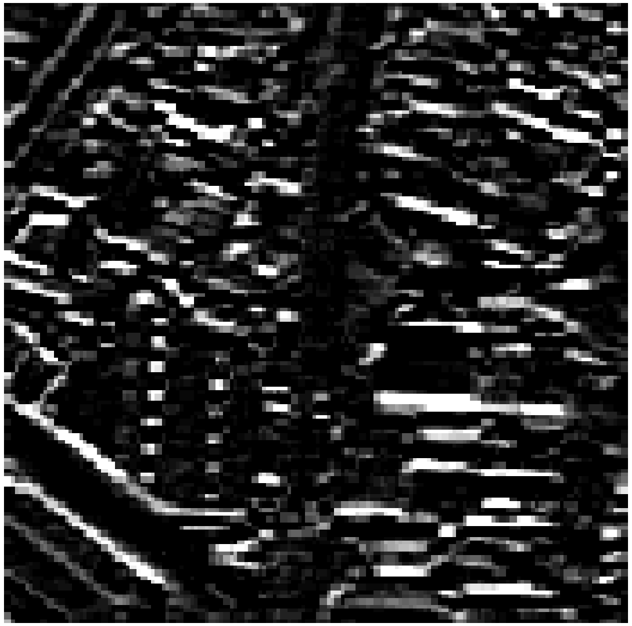
- A convolução de uma janela e a imagem também é útil para salientar determinadas linhas ou bordas. Por exemplo, as técnicas de filtragem permitem salientar as bordas ou linhas que ocorrem numa determinada direção, fazendo a diferença dos valores na janela considerando sua posição em relação ao pixel central da janela. A seguir são mostrados alguns exemplos destes filtros.



Exemplo: bordas horizontais

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & -2 & 1 \\ -1 & -1 & -1 \end{bmatrix}$$

Norte



- Os contrates na direção norte são salientados.
- Algumas linhas diagonais também são salientadas, pois possuem uma componente norte forte.

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & -2 & 1 \\ -1 & -1 & -1 \end{bmatrix}$$

Norte

$$\begin{bmatrix} 1 & 1 & 1 \\ -1 & -2 & 1 \\ -1 & -1 & 1 \end{bmatrix}$$

Nordeste

$$\begin{bmatrix} -1 & -1 & -1 \\ 1 & -2 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

Sul

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & -2 & -1 \\ 1 & -1 & -1 \end{bmatrix}$$

Noroeste

$$\begin{bmatrix} -1 & 1 & 1 \\ -1 & -2 & 1 \\ -1 & 1 & 1 \end{bmatrix}$$

Leste

$$\begin{bmatrix} -1 & -1 & 1 \\ -1 & -2 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

Sudeste

$$\begin{bmatrix} 1 & 1 & -1 \\ 1 & -2 & -1 \\ 1 & 1 & -1 \end{bmatrix}$$

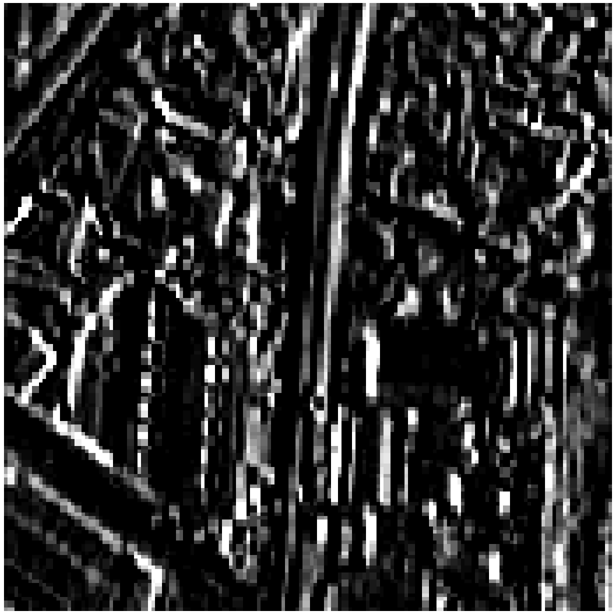
Oeste

$$\begin{bmatrix} 1 & -1 & -1 \\ 1 & -2 & -1 \\ 1 & 1 & 1 \end{bmatrix}$$

Sudoeste

$$\begin{array}{rrr} -1 & 1 & 1 \\ -1 & -2 & 1 \\ -1 & 1 & 1 \end{array}$$

leste



•Linhas horizontais, mas todas?

Filtros não lineares

- Resultam da análise da vizinhança em torno do pixel, mas neste caso seu funcionamento não pode ser representado usando a forma geral da convolução.
- **O filtro de moda** (valor mais frequente) : Usado para suavizar imagens, especialmente temáticas, pois o novo valor atribuído ao pixel central corresponde ao valor mais frequente da vizinhança e por este motivo é igual a pelo menos um dos pixels vizinhos.
- **O filtro de mediana** (valor central) : o novo valor corresponde ao valor central após ordenar os valores de forma crescente.

1	1	1	1	2	1	1	0
1	1	2	1	3	1	1	1
1	1	3	1	2	1	1	0
1	1	3	1	4	1	1	1
9	7	8	8	8	9	9	8
9	9	8	8	9	9	9	8
9	9	9	8	9	8	9	9
9	8	9	8	8	9	9	9

Serie=[3 1 2 3 1 4 8 8 8]
Moda: 8 (mais frequente)

Mediana:
[1 1 2 3 3 4 8 8 8] = 3

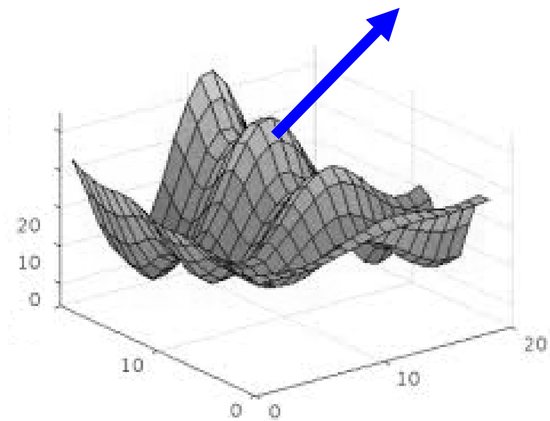
Mediana

Filtro da mediana: O valor resultante é a mediana da vizinhança. Este filtro introduz um certo grau de suavização na imagem resultante, do que decorre perda de detalhe. A diferença em relação ao filtro passa baixas é que as bordas não são degradadas em extremo, pois os valores originais são preservados.

Filtros de Gradiente

O gradiente de uma superfície descreve sua inclinação no local especificado e é um vetor, que aponta para fora da superfície.

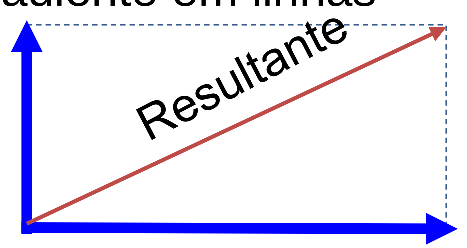
O gradiente pode ser calculado a partir de suas duas componentes (Norte e Leste), ou seja, a derivada parcial da função da superfície em relação a linhas e colunas.



$$G(x,y) = \begin{bmatrix} \frac{\delta F(x,y)}{\delta(x)} \\ \frac{\delta F(x,y)}{\delta(y)} \end{bmatrix}$$

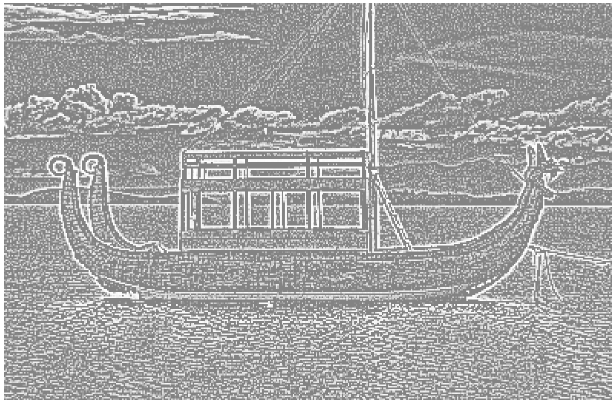
Intensidade do Gradiente

$$I(x,y) = \sqrt{\frac{\delta F(x,y)}{\delta(x)}^2 + \frac{\delta F(x,y)}{\delta(y)}^2}$$



No Para calcular o Gradiente:

- Estima-se o gradiente em X
- Estima-se o gradiente em Y (Y perpendicular a X)
- Calcula-se a resultante da soma destes dois vetores.
- O pixel recebe um valor proporcional à magnitude do gradiente.



No processamento de imagens, pode-se assumir que a variação dos valores digitais se assemelha a uma superfície, similar a um Modelo Digital do Terreno (MDT).

Logo, torna-se possível calcular o gradiente para qualquer pixel, analisando a variação dos valores em sua vizinhança.

Para isto:

- Estima-se o gradiente em X
- Estima-se o gradiente em Y (Y perpendicular a X)
- Calcula-se a resultante da soma destes dois vetores.
- O pixel recebe um valor proporcional à magnitude do gradiente.

A diferença entre os filtros de gradiente radica na maneira de estimar as duas derivadas parciais.

Ex: filtros de Roberts:

$$\boxed{\begin{matrix} 1 & 0 \\ 0 & -1 \end{matrix}} \qquad \boxed{\begin{matrix} 0 & 1 \\ -1 & 0 \end{matrix}}$$

δF(x,y)/δ(x)　　δF(x,y)/δ(y)

Magnitude:

$$I(x,y) = \sqrt{\frac{\delta F(x,y)}{\delta(x)}^2 + \frac{\delta F(x,y)}{\delta(y)}^2}$$

$$\frac{\delta F(x,y)}{\delta(x)} = F(x-1, y-1) - F(x,y)$$

$$\frac{\delta F(x,y)}{\delta(y)} = F(x-1, y+1) - F(x,y)$$

- Ex: filtros de Prewitt

$$\begin{array}{ccc} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{array} \qquad \delta F(x,y)/\delta(x)$$

$$\begin{array}{ccc} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{array} \qquad \delta F(x,y)/\delta(y)$$

Magnitude:

$$I(x,y) = \sqrt{\frac{\delta F(x,y)}{\delta(x)}^2 + \frac{\delta F(x,y)}{\delta(y)}^2}$$

- Ex: filtros de Sobel

$$\boxed{\begin{matrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{matrix}}$$

δF(x,y)/δ(x)

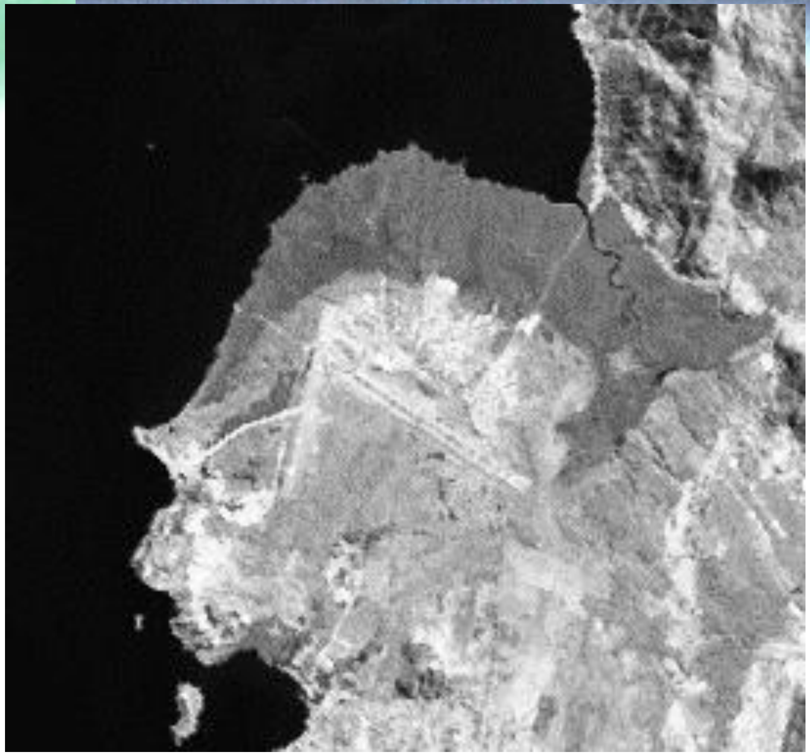
$$\boxed{\begin{matrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{matrix}}$$

δF(x,y)/δ(y)

Magnitude:

$$I(x,y) = \sqrt{\frac{\delta F(x,y)}{\delta(x)}^2 + \frac{\delta F(x,y)}{\delta(y)}^2}$$

Ex: filtro de SOBEL



Áreas uniformes: baixo gradiente
áreas de fronteira são salientadas, bem como feições lineares.

- Como seria um programa de filtro?

- Aula prática

dtype = None

```
img = cv2.imread(nome)   # ler imagem
n,m,nb = img.shape
I=img[:,:,2]   # separe uma banda
J=np.zeros([n,m], dtype='float') # crie uma saida
for i in range(n): # varrerlinahs e colunas
  for j in range(m):
      s=0           # inicia uma soma
      for dl in range(3):   # varia vizinhança
               for coluna in range(3):
                      s=s+I[i+dl-1, j+dc-1]  # acumula a soma
        s=s/9    # calcula a média
        J(i,j)=s # atribuir a saída (cuidado! Devemos arredondar
```

1	1	1	1	2	1	1	0
1	1	2	1	3	1	1	1
1	1	3	9	2	1	1	0
1	1	3	1	2	1	1	1
9	7	8	9	8	9	9	8
9	9	8	8	9	9	9	8
9	9	9	8	9	8	9	9
9	8	9	8	8	9	9	9

Isto não funciona para os pixels nas bordas da imagem, pois o vizinho anterior à primeira linha não existe!

Obs: o que ocorre se somarmos os oito valores uint8 ?

Restringir pixels a serem processados em função do tamanho da vizinhança.

```
I=img[:,:,2]
J=np.zeros((n,m),dtype = np.uint8 )  # criamos uma 
variável vazia em uint8
for i in range(1,n-1):  # valores realmente possíveis
  for j in range(1,m-1):
    s=0
    for dl in range(3):
      for dc in range(3):
        v=float( I[i+dl-1, j+dc-1]) # converter a float 
para somar mais de 255
        s=s + v
    s=np.round(s/9)
    J[i,j]=np.uint8(s)
```

1	1	1	1	2	1	1	0
1	1	2	1	3	1	1	1
1	1	3	9	2	1	1	0
1	1	3	1	2	1	1	1
9	7	8	9	8	9	9	8
9	9	8	8	9	9	9	8
9	9	9	8	9	8	9	9
9	8	9	8	8	9	9	9

- No ambiente Google Colab

- Desenvolva um filtro de média (3x3), depois um (5x5)
- Um filtro passa-altas...

Programa (Passa-baixas)

1	1	1
1	1	1
1	1	1

1	1	1	1	2	1	1	0
1	1	2	1	3	1	1	1
1	1	3	9	2	1	1	0
1	1	3	1	2	1	1	1
9	7	8	9	8	9	9	8
9	9	8	8	9	9	9	8
9	9	9	8	9	8	9	9
9	8	9	8	8	9	9	9

Elabore um programa de filtro de médias (passa-baixas) no que o usuário possa variar o tamanho da janela móvel)

dim=3,5,7,..?

Vizinhos antes e depois

lado=(dim-1)/2

Verifique para dim=3, ou 5

No 3x3, a varredura não pode ser feita para o primeiro pixel, devemos começar no elemento (lado+1)

e não podemos terminar na ultima linha (n) mas devemos terminar em (n-[lado-1])

E no caso 5x5 ?

1	1	1	1	2	1	1	0
1	1	2	1	3	1	1	1
1	1	3	9	2	1	1	0
1	1	3	1	2	1	1	1
9	7	8	9	8	9	9	8
9	9	8	8	9	9	9	8
9	9	9	8	9	8	9	9
9	8	9	8	8	9	9	9

1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1